# num-rows

mysqli_stmt::$num_rows

mysqli_stmt::num_rows

mysqli_stmt_num_rows

Returns the number of rows fetched from the server

ENTITY

ENTITY

int

string

mysqli_stmt-ENTITYnum_rows

public

int

string

mysqli_stmt::num_rows

ENTITY

int

string

mysqli_stmt_num_rows

mysqli_stmt

statement

Returns the number of rows buffered in the statement. This function will only work after `mysqli_stmt_store_result` is called to buffer the entire result set in the statement handle.

This function returns `0` unless all rows have been fetched from the server.

ENTITY

ENTITY

An ENTITY representing the number of buffered rows. Returns `0` in unbuffered mode unless all rows have been fetched from the server.

ENTITY

ENTITY

ENTITY

```
<?php

mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);
$mysqli = new mysqli("localhost", "my_user", "my_password", "world");

$query = "SELECT Name, CountryCode FROM City ORDER BY Name LIMIT 20";
$stmt = $mysqli->prepare($query);
$stmt->execute();

/* store the result in an internal buffer */
$stmt->store_result();

printf("Number of rows: %d.\n", $stmt->num_rows);
```

ENTITY

```
<?php

mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);
$link = mysqli_connect("localhost", "my_user", "my_password", "world");
```

```
$query = "SELECT Name, CountryCode FROM City ORDER BY Name LIMIT 20";
$stmt = mysqli_prepare($link, $query);
mysqli_stmt_execute($stmt);

/* store the result in an internal buffer */
mysqli_stmt_store_result($stmt);

printf("Number of rows: %d.\n", mysqli_stmt_num_rows($stmt));
```

ENTITY

```
Number of rows: 20.
```

ENTITY

`mysqli_stmt_store_result`, `mysqli_stmt_affected_rows`, `mysqli_prepare`

---

□□□ #2
□ □□□ □□□ 2026-10-31 17:59:30 UTC
□ □□□ □□□ 2026-10-31 17:59:32 UTC